

# КОМПЬЮТЕРНАЯ

# ИНЖЕНЕРИЯ И

# ТЕХНИЧЕСКАЯ

# ДИАГНОСТИКА



УДК 681.325:519.613

## ПРОЕКТИРОВАНИЕ ТЕСТОВ ДЛЯ СТРУКТУРНО-ФУНКЦИОНАЛЬНЫХ МОДЕЛЕЙ ЦИФРОВЫХ СХЕМ

*ХАХАНОВ В.И., СЫСЕНКО И.Ю.,  
АБУ ЗАНУНЕХ ХАЛИЛЬ И.М.*

Предлагается метод генерации тестов для структурно-функциональных моделей цифровых устройств на основе активизации одномерных логических путей при использовании кубических покрытий (КП) функциональных элементов. Метод позволяет проектировать проверяющие тесты для всех неисправностей существенных линий схемы.

### 1. Введение

Регулярные методы генерации тестов ориентированы на проверку исправности или неисправности наперед заданного класса (чаще всего это — одиночные константные неисправности (ОКН)). Известные алгоритмы построения тестов [1,2] имеют определенные недостатки, связанные с формализацией выполнения процедур прямой и обратной импликаций, что в общем случае не гарантировало получение полного проверяющего теста относительно ОКН для произвольных цифровых структур. Многочисленные научные исследования, обобщенные в [1-3], посвящены решению задач проектирования проверяющих последовательностей для схем со сходящимися разветвлениями, анализ которых традиционно считается трудноформализуемым. Предлагается метод двухфреймовой активизации одномерных логических путей и его теоретическое обоснование в двухтактном алфавите кубического исчисления с последующим применением алгебры Кантора для метода активизации. Ниже представлена задача генерации тестов как одна из проблем логического анализа цифровых схем.

### 2. Формализация задач логического анализа

Моделирование — процесс определения неизвестных компонентов тетрады  $\langle M, D, T, R \rangle$ :  $\langle$  модель, дефекты, стимулы, реакция  $\rangle$  на основе применения правил логического вывода.

Обобщенное уравнение анализа модели цифрового изделия при наличии или отсутствии неисправностей  $M = R \vee DR \vee TR \vee DTR = R(1 \vee D \vee T \vee DT)$  формализует постановку следующих задач.

1. Определение модели устройства в виде совокупности всех его возможных реакций на функциональные входные наборы:

$$M = f^{1,1}(T, D, R)|_{D=\emptyset; T=\emptyset}.$$

2. Создание модели цифрового устройства в виде совокупности всех его возможных реакций на функциональные входные наборы при наличии неисправностей заданного класса (словарь неисправностей):

$$M = f^{1,2}(T, D, R)|_{T=\emptyset}.$$

3. Построение модели цифрового устройства в виде входных наборов и соответствующих им реакций — таблица истинности (переходов):

$$M = f^{1,3}(T, D, R)|_{D=\emptyset}.$$

4. Создание модели устройства в виде входных тестовых наборов и соответствующих им реакций при наличии заданных неисправностей — таблица функций неисправностей:

$$M = f^{1,4}(T, D, R).$$

5. Определение множества потенциальных, фактических или проверяемых дефектов в цифровом изделии:

$$D = M \vee MT \vee MR \vee MTR = M(1 \vee T \vee R \vee TR).$$

6. Генерация списков неисправностей цифрового изделия по его модели в виде константных или функциональных дефектов:

$$D = f^{2,1}(M, T, R)|_{T=\emptyset; R=\emptyset}.$$

7. Моделирование заданных неисправностей цифрового изделия на входных наборах в целях определения качества теста:

$$D = f^{2,2}(M, T, R)|_{R=\emptyset}.$$

8. Определение фактических неисправностей цифрового изделия на функциональных наборах по экспериментальным реакциям — обратное проследивание дефектов:

$$D = f^{2,3}(M, T, R)|_{T=\emptyset}.$$

9. Определение фактических неисправностей цифрового изделия на тестовых наборах по экспериментальным реакциям — безусловное диагностирование:

$$D = f^{2,4}(M, T, R).$$

10. Построение теста верификации модели, проверки исправности, обнаружения заданных неисправностей:

$$T = M \vee MD \vee MR \vee MDR = M(1 \vee D \vee R \vee DR).$$

11. Проектирование теста (проверки исправности) цифрового изделия по его модели:

$$T = f^{3,1}(M, D, R)|_{D=\emptyset; R=\emptyset}.$$

12. Проектирование теста (проверяющего, диагностирования) цифрового изделия по его модели и заданным неисправностям:

$$T = f^{3,2}(M, D, R)|_{R=\emptyset}.$$

13. Проектирование теста (проверки исправности) цифрового изделия по его модели исправного поведения и заданным реакциям — обратная импликация:

$$T = f^{3,3}(M, D, R)|_{D=\emptyset}.$$

14. Проектирование теста (проверяющего, диагностирования) — определение состояний входных переменных цифрового изделия по его модели, заданным дефектам и реакциям — обратная импликация:

$$T = f^{3,4}(M, D, R).$$

15. Определение реакции цифрового изделия — прямая импликация:

$$R = M \vee MD \vee MT \vee MDT = M(1 \vee D \vee T \vee DT).$$

16. Определение состояний невходных линий цифрового изделия как реакции модели на входные последовательности:

$$R = f^{4,1}(M, D, T) |_{D=\emptyset; T=\emptyset}.$$

17. Определение состояний невходных линий цифрового изделия как реакции модели на входные последовательности при наличии заданных неисправностей:

$$R = f^{4,2}(M, D, T) |_{T=\emptyset}.$$

18. Определение состояний невходных линий цифрового изделия как реакции модели на входные тестовые последовательности:

$$R = f^{4,3}(M, D, T) |_{D=\emptyset}.$$

19. Определение состояний невходных линий цифрового изделия как реакции модели на входные тестовые последовательности при наличии заданных дефектов – моделирование неисправностей заданного класса:

$$R = f^{4,4}(M, D, T) |.$$

В формулировках проблем  $f^{i,j}$  ( $i=1,4; j=1,4$ ) есть система логических преобразований (методы и алгоритмы), предназначенная для рационального решения конкретной задачи анализа или проектирования.

Задача генерации тестов, рассматриваемая далее, формализуется в правилах 14.

### 3. Аппарат двухтактного кубического исчисления для генерации тестов

Для задания моделей функциональных элементов и их анализа будем использовать их табличное представление, как наиболее технологичную форму восприятия информации пользователем и компьютером, которая основывается на кубическом исчислении – математическом аппарате компактного описания цифровых структур.

Основной недостаток табличного представления конечных автоматов – размерность – может быть сведен к минимуму введением избыточности в алфавит кодирования состояний булевых переменных. Алгебра Кантора есть алфавит описания автоматных переменных в одном временном фрейме:

$$A^1 = \{0, 1, X = \{0, 1\}, \emptyset\}.$$

Основным компонентом двухтактного кубического исчисления (ДКИ) является алфавит [3]:  $A^x = \{Q=00, E=01, H=10, J=11, O=\{Q, H\}, I=\{E, J\}, A=\{Q, E\}, B=\{H, J\}, S=\{Q, J\}, P=\{E, H\}, C=\{E, H, J\}, F=\{Q, H, J\}, L=\{Q, E, J\}, V=\{Q, E, H\}, Y=\{Q, E, H, J\}, U\}$ , который задает все возможные переходы двухтактной автоматной переменной в моменты  $(t-1, t)$  универсумом примитивов  $Y = \{00, 01, 10, 11\}$ . Символ  $U$  обозначает пустое множество  $\emptyset$  и служит для замыкания алфавита относительно теоретико-множественных операций, для его связи с математическими структурами более высокой иерархии и построения систем доказательств.

Объединяющим форматом для  $A^x$  и  $A^1$  служат два временных фрейма  $\langle t-1, t \rangle$ . Приведение одноктактных символов  $a_i^1 \in A^1$  к двухтактной форме выполняется с помощью символа  $Z$ , который эквивалентен пустому множеству  $U$  в одном автоматном такте, но "чуть-чуть" полнее последнего, и задает описание переменной в состоянии высокого импеданса или высокого выходного сопротивления. С по-

мощью буквы  $Z$  формат одноктактных символов принимает вид:  $a_i^1 \langle Z, a \rangle$ , где  $a$  – одноктактные булевы состояния из множества  $A = \{0, 1, X = \{0, 1\}, Z\}$ , абстрагированные от времени.

По определению на каждом множестве  $A^1, A^X$  выполняются все аксиомы и тождества абстрактной математической решетки с дополнениями, такие как ассоциативность, коммутативность, дистрибутивность, идемпотентность, действия с универсумом и пустым множеством, двойного дополнения, правила де-Моргана, поглощения.

### 4. Проектирование двухтактных КП активизации

Основной недостаток D-алгоритма и других методов активизации логических путей заключается в итеративном характере получения многомерного пути активизации при наличии сходящихся разветвлений в целях очувствления некоторого входа относительно наблюдаемых выходов, если такое возможно. Другое решение данной задачи связано с использованием ДКИ. Очувствление входной переменной с помощью построения одно- или многомерных путей активизации осуществляется за одну итерацию, которая включает прямую импликацию символов активизации и обратную импликацию символов обеспечения. Рассмотрим теоретическое обоснование кубического (К-) алгоритма.

**Лемма 1.** Покрытие активизации примитивного элемента (ПЭ) из базиса  $W = \{И, ИЛИ, И-НЕ, ИЛИ-НЕ\}$  записывается на алфавите  $A^9 \subset A^X$ .

Поскольку кубическое покрытие любой комбинационной схемы с учетом возможной минимизации представляется значащими символами одноктактного кубического исчисления  $A^1 = \{0, 1, X\}$ , то конкатенация всех возможных упорядоченных пар символов создает подмножество в виде:

*	0	1	X
0	Q	E	A
1	H	J	B
X	O	I	Y

(1)

Следовательно, другие символы для упомянутых логических элементов появиться не могут, поэтому определим усеченные операции пересечения и объединения, представленные в табл. 1, где в качестве исходных используются 9 символов, полученных в (1).

Определение 1. Покрытие активизации называется полным, если оно содержит все возможные решения, формирующие на выходе ПЭ символы E, H.

Таблица 1

		Пересечение								
		Q	E	H	J	O	I	A	B	Y
Q	Q	Q	U	U	U	Q	U	Q	U	Q
E	E	U	E	U	U	U	E	E	U	E
H	H	U	U	H	U	H	U	U	H	H
J	J	U	U	U	J	U	J	U	J	J
O	O	Q	U	H	U	O	U	Q	H	O
I	I	U	E	U	J	U	I	E	J	I
A	A	Q	E	U	U	Q	E	A	U	A
B	B	U	U	H	J	H	J	U	B	B
Y	Y	Q	E	H	J	O	I	A	B	Y

		Объединение								
		Q	E	H	J	O	I	A	B	Y
Q	Q	Q	A	O	S	O	L	A	F	Y
E	E	A	E	P	I	V	I	A	C	Y
H	H	O	P	H	B	O	C	V	B	Y
J	J	S	I	B	J	F	I	L	B	Y
O	O	O	V	O	F	O	Y	V	F	Y
I	I	L	I	C	I	Y	I	L	C	Y
A	A	A	A	V	L	V	L	A	Y	Y
B	B	F	C	B	B	F	C	Y	B	Y
Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y

E(H)-покрытие обеспечивает на выходе изменения из 0(1) в 1(0). Покрытие обеспечения называется полным, если оно содержит все возможные решения, формирующие на выходе ПЭ символы O, I, A, B, Q, J. Объединение упомянутых полных покрытий называется полным двухтактным покрытием, которое обеспечивает прямую и обратную импликации символов {Q, E, H, J, O, I, A, B}.

**Теорема 1.** Полное покрытие активизации элемента из базиса W содержит 2n кубов и может быть получено конкатенацией в обоих направлениях нулевых и единичных кубов.

Поскольку элемент из базиса W имеет регулярное покрытие, содержащее один нулевой (единичный) и n единичных (нулевых) кубов, то число сконкатенированных в одном направлении пар всегда будет равно количеству входов. Но является ли полученное таким образом КП полным, в том смысле, что оно содержит кубы активизации кратности 1, 2, ..., n? Для ответа на данный вопрос рассмотрим КП элемента 3И-НЕ:

$$C = \begin{Bmatrix} X_1 & X_2 & X_3 & Y \\ 0 & X & X & 1 \\ X & 0 & X & 1 \\ X & X & 0 & 1 \\ 1 & 1 & 1 & 0 \end{Bmatrix}. \quad (2)$$

Сцепление по правилам упомянутой выше \*-операции (1) первых трех кубов с последним дает результат:

$$C = \begin{Bmatrix} X_1 & X_2 & X_3 & Y \\ E & I & I & H \\ I & E & I & H \\ I & I & E & H \end{Bmatrix}. \quad (3)$$

Кроме того, известное полное покрытие активизации [3] для упомянутого примитива имеет 7 кубов, которые предназначены для реализации активизации каждой переменной в отдельности, всех возможных сочетаний по два активных входа на трех линиях, а также одновременной активизации трех переменных:

$$C = \begin{Bmatrix} X_1 & X_2 & X_3 & Y \\ D & 1 & 1 & N \\ 1 & D & 1 & N \\ 1 & 1 & D & N \\ D & D & 1 & N \\ D & 1 & D & N \\ 1 & D & D & N \\ D & D & D & N \end{Bmatrix}. \quad (4)$$

Выполнив замену знаков 1, D, N на соответствующие им символы двухтактного алфавита, получим:

$$C = \begin{Bmatrix} X_1 & X_2 & X_3 & Y \\ H & J & J & E \\ J & H & J & E \\ J & J & H & E \\ H & H & J & E \\ H & J & H & E \\ J & H & H & E \\ H & H & H & E \end{Bmatrix}. \quad (5)$$

Кубы 1, 2, 3 могут быть склеены с использованием операции объединения со строками 4, 6, 5 соответственно, а куб 7 можно склеить с векторами 4, 5, 6. Получается покрытие:

$$C = \begin{Bmatrix} X_1 & X_2 & X_3 & Y \\ H & B & J & E \\ J & H & B & E \\ B & J & H & E \\ H & H & H & E \\ H & B & H & E \\ B & H & H & E \end{Bmatrix}. \quad (6)$$

Вторая итерация минимизации наборов 1, 2, 3 с кубами 5, 4, 6 соответственно дает окончательный результат в виде E-покрытия:

$$C = \begin{Bmatrix} X_1 & X_2 & X_3 & Y \\ H & B & B & E \\ B & H & B & E \\ B & B & H & E \end{Bmatrix}, \quad (7)$$

инверсия во времени которого с использованием одноместной операции

$$\frac{C_{ij}}{\bar{C}_{ij}} = \begin{Bmatrix} Q & E & H & J & O & I & A & B & S & P & C & F & L & V & Y \\ Q & H & E & J & A & B & O & I & S & P & C & F & L & V & Y \end{Bmatrix} \quad (8)$$

определяет H-покрытие активизации:

$$C = \begin{Bmatrix} X_1 & X_2 & X_3 & Y \\ E & I & I & H \\ I & E & I & H \\ I & I & E & H \end{Bmatrix}. \quad (9)$$

Оба упомянутых покрытия и составляют полное КП активизации.

По индукции данное доказательство может быть распространено на n-входные логические элементы, которые будут представлены 2n кубами активизации, включающими векторы мерности 1, 2, 3, ..., n.

Поскольку для комбинационных схем направленные активизации являются несущественным для проверки неисправностей линий, то в целях экономии памяти можно иметь покрытие очувствления линий в одном направлении, которое при необходимости следует трансформировать в покрытие активизации обратного знака с помощью операции временной инверсии (8).

Определение 2. Тривиальным называется куб, имеющий по входным и выходным координатам символы X.

**Теорема 2.** Полное двухтактное покрытие обеспечения содержит 3(n+1) кубов для n-входового элемента базиса W.

Упомянутое покрытие должно обеспечивать обратную импликацию всех девяти символов, получаемых при выполнении конкатенации (1): {Q, J, E, H, O, I, A, B, Y}. Символ Y не нуждается в обратной импликации, а для обеспечения {E, H} следует использовать ранее определенные покрытия активизации. Остаются еще пять символов. Но поскольку пары символов (O, I), (A, B) являются взаимно инверсными во времени, то для каждого логического примитива проведем доказательство получения одной из пар символов, например, (O, I). Символы можно представить упорядоченной парой: O=(X0), I=(X1). Значения 0 и 1 в момент t определяются состояниями выходов однократного кубического покрытия (2). Состояние выхода ПЭ в момент t-1 равно X, которое может быть получено следующими комбинациями символов X и 1 (0 - для элемента ИЛИ), число которых определяется выражением 2<sup>n</sup>-1:

$$C = \begin{Bmatrix} X_1 & X_2 & X_3 & Y \\ 1 & X & X & X \\ X & 1 & X & X \\ J & X & 1 & X \\ 1 & 1 & X & X \\ 1 & X & 1 & X \\ X & 1 & 1 & X \\ X & X & X & X \end{Bmatrix}. \quad (10)$$

Приведенные 7 кубов минимизируются, превращаясь в один тривиальный куб XXXX, для которого любое решение при выполнении прямой или обратной импликации в силу своего определения является непротиворечивым. Отсюда следует, что двухтактное покрытие обеспечения для символов {O, I} составляется конкатенацией всех кубов однократного КП с тривиальным кубом, который содержит все возможные значения входов для X-состояния выхода, с получением для элемента 3И-НЕ строк:

$$C = \begin{Bmatrix} X_1 & X_2 & X_3 & Y \\ O & Y & Y & I \\ Y & O & Y & I \\ Y & Y & O & I \\ I & I & I & O \end{Bmatrix}. \quad (11)$$

Кубы A, B-покрытия получаются применением операции временной инверсии к координатам приведенного выше КП, что дает результат:

$$C = \begin{Bmatrix} X_1 & X_2 & X_3 & Y \\ A & Y & Y & B \\ Y & A & Y & B \\ Y & Y & A & B \\ B & B & B & A \end{Bmatrix}. \quad (12)$$

Кубы обеспечения символов {Q, J} получаются из КП ПЭ путем конкатенации каждого куба с самой собой, что дает результат:

$$C = \begin{Bmatrix} X_1 & X_2 & X_3 & Y \\ Q & Y & Y & J \\ Y & Q & Y & J \\ Y & Y & Q & J \\ J & J & J & Q \end{Bmatrix}. \quad (13)$$

По индукции для любого элемента базиса W всегда можно выполнить аналогичные рассуждения. Поскольку тривиальный куб всегда один, а мощность однократного покрытия элемента базиса W равна  $n+1$ , то число кубов полного покрытия обеспечения символов {O, I, A, B, Q, J} равно  $3(n+1)$ , что и требовалось доказать.

### 5. Процедуры кубического алгоритма

**Лемма 2.** Выполнение прямой импликации в комбинационной схеме не порождает новых символов, отличных от {O, I, A, B} или их подмножеств, нуждающихся в доопределении.

Покрытия активизации элементов базиса W не имеют других символов обеспечения, кроме {O, I, A, B}. В результате выполнения прямой импликации осуществляется лишь операция пересечения по координатам кубов активизации, которая не может увеличить мощность символа, а может только уменьшить, порождая значения {E, H, Q, J}.

**Следствие 1.** При выполнении прямой импликации не могут появиться символы алфавита  $A^X$ , мощности 3, определяемые подмножеством: {C, F, L, V}.

**Лемма 3.** Обратная импликация не увеличивает количество неопределенностей при ее выполнении в целях доопределения символов обеспечения.

При обработке очередного элемента, выход которого определен символами подмножества {O, I, A, B}, возможны два случая. В первом входы ПЭ не являются линиями разветвления, а каждый из них соединен только с одним выходом другого примитива. Тогда состояния линий, соответствующих входам рассматриваемого ПЭ, будут определяться значениями координат КП обеспечения, которые могут быть заданы символами {O, I, A, B}. Во втором случае один или несколько входов элемента принадлежат линиям разветвления. Тогда общее непротиворечивое решение для них будет определяться пересечением. Поскольку упомянутая координатная операция может только уменьшить мощность результирующего символа, то символы подмножества {C, F, L, V} не могут появиться при выполнении обратной импликации.

**Следствие 2.** Реализация кубического алгоритма для комбинационных схем вентильного базиса W выполняется на подмножестве двухтактного алфавита  $A^9$ , порожденного сокращенной операцией конкатенации (1) при использовании полных двухтактных покрытий активизации и обеспечения.

**Следствие 3.** Для выполнения обратной импликации двухуровневых схем необходимо и достаточно использования {O, I, A, B}-покрытий.

**Следствие 4.** Для выполнения обратной импликации неизбыточных и двухуровневых схем покрытия активизации не используются.

**Определение 3.** Выбор одного из возможных решений при выполнении обратной импликации через ПЭ, удовлетворяющих исходному состоянию выхода, будем называть одновариантным анализом. Использование всех возможных решений входных комбинаций, не противоречащих требуемому состоянию выхода, есть многовариантная обратная импликация.

**Теорема 3.** Кубический алгоритм, использующий полные двухтактные покрытия активизации и обеспечения, выполняет построение пути активизации минимальной мерности за одну итерацию процедур прямой и обратной многовариантной импликации или определяет невозможность получения такого пути.

Построение одномерного пути активизации при выполнении прямой фазы определяет активные линии и условия очувствления на входах ПЭ, которые формируют единственный вектор активности, поскольку прямая импликация есть одновариантная задача. Противоречия в данной процедуре возникнуть не могут, если элементы имеют полные покрытия активизации, а это действительно так по условию теоремы 1. Обратная фаза основывается на использовании полных двухтактных покрытий обеспечения согласно теореме 2. Значит, при обработке очередного ПЭ, что является многовариантной задачей (поскольку в общем случае существует более одного куба для получения некоторого двухтактного состояния выхода), решение обязательно будет получено по входам примитива с учетом, что полное покрытие содержит все возможные кубы для определения состояний {0, 1, X} на выходах элемента. Если в силу структуры гальванических связей, образующих разветвления на входах ПЭ, возникнет пустое пересечение на одной или нескольких входных линиях между всеми кубами покрытия обеспечения и текущими состояниями вектора активизации  $D_i$ , делается вывод о противоречивости текущего

решения  $D_i$ . Если для всех  $D_i$  отсутствует положительное пересечение с кубами анализируемого покрытия, тест активизации для выбранного логического пути не существует.

При построении тестов для комбинационной схемы (КС) будем использовать стратегию построения минимального множества одномерных путей активизации, покрывающих все существенные эквипотенциальные линии устройства. Любой активный путь всегда начинается с постановки символа очувствления на очередном внешнем входе. Далее выбирается один из элементов, которому принадлежит данный вход, что обеспечивает тенденцию к желанию построить одномерный путь активизации. Первое пересечение символа  $E$  или  $H$  с покрытием активности выбранного ПЭ формирует вектор активизации, который после обработки примитива будет иметь на линии, соответствующей выходу ПЭ, символ из множества  $\{E, H\}$ . Далее вектор активизации пересекается с кубами покрытия элемента, входы которого соединены с уже активным выходом. Такая процедура заканчивается при достижении внешних выходов схемы.

**Лемма 4.** Для выполнения прямой или обратной импликации используются только операции пересечения и временной инверсии.

В соответствии со стратегией проектирования одномерных путей после определения на очередном входе (линии) вектора активизации символа из множества  $\{E, H\}$  все операции прямой фазы сводятся к пересечению вектора  $D_i$  с кубами очередного ПЭ. В случае пустого решения выполняется временная инверсия кубов активности, и пересечение повторяется. В обратной фазе наблюдается аналогичная ситуация с той лишь разницей, что вектор  $D_i$  может размножаться, благодаря многовариантности обратной импликации.

**Лемма 5.** Двухтактные покрытия активизации используются для выполнения прямой и обратной импликации.

В соответствии со стратегией проектирования одномерных путей вначале всегда определяется символ активизации на входе очередного элемента, после чего выполняется пересечение кубов покрытия с вектором активизации в целях определения очередной активной линии, относящейся к выходу ПЭ. Кроме того, при выполнении обратной фазы на линии сходящегося разветвления возможно возникновение символов активизации  $H, E$  при пересечении символов  $O$  и  $B$ ;  $A$  и  $I$ , что является штатной ситуацией (рис. 1). При построении активного пути  $a$ - $c$ - $d$  вход  $b$  элементов 1 и 2 требовали условий  $I, A$  соответственно, что не является противоречием и при пересечении дает символ  $E$ .

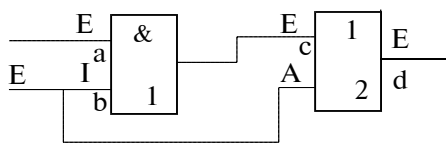


Рис. 1. Порождение активизации на обратной фазе

**Теорема 4.** Двухтактное полное покрытие  $n$ -входного элемента из множества  $W$  для выполнения кубического алгоритма содержит  $5n+3$  куба.

Мощность покрытия для выполнения процедуры обеспечения символов  $\{O, I\}$  равна  $n+1$ . Аналогичную оценку имеют покрытия, создающие на выходе ПЭ

символы  $\{A, B\}$  и  $\{Q, J\}$  (теорема 2). Мощность покрытия активизации одного направления (теорема 1), например, для обеспечения на выходе символа  $E$  равна  $n$ . Дополняющее покрытие типа  $H$  также содержит  $n$  кубов. Суммируя все упомянутое, получаем оценку  $5n+3$ , что и требовалось доказать.

**Теорема 5.** Для схемы, соответствующей СДНФ, существует множество одномерных путей активизации, покрывающих все существенные переменные.

Совершенная дизъюнктивная нормальная форма содержит количество конъюнктивных термов, равное числу единиц таблицы истинности. Каждый терм может содержать существенные и несущественные переменные. Последние определяются невозможностью активизировать значение выхода примитива, соответствующего рассматриваемому терму. Это означает, что два соседних входных состояния формируют единичные (нулевые) значения выходов рассматриваемой функции. Кроме того, наличие в карте состояний функции (Карно) двух соседних единиц есть не только условие невозможности активизации выхода при изменении входной переменной, определяющей координатное различие единичного состояния функции, но и условие несущественности упомянутой входной переменной. Но если две соседние по Хеммингу координаты в карте Карно имеют различные значения, то изменение входных слов, соответствующих упомянутым координатам, есть необходимое и достаточное условие для активизации функции в целом, поскольку не существует другого простого терма СДНФ, который на заданной паре входных наборов имел бы активность. Верно также и то, что не существует такого терма, который смог бы компенсировать выходную активность примитива в силу детерминизма конечного комбинационного автомата, который всегда имеет нулевую или единичную реакцию на двоичное входное слово. Таким образом, если переменная в терме существенна, ее активизация выполняется построением одномерного логического пути, который всегда существует. Следовательно, для всех существенных переменных термов СДНФ существуют одномерные пути активизации.

## 6. Особенности построения тестов для избыточных схем

**Следствие 5.** Если не существует пары входных наборов, активизирующих переменную в терме СДНФ одномерным путем, она является несущественной, но может быть активизирована многомерно.

В самом деле, если изменение на одном входе не приводит к изменению выхода функции, то соседние клетки в карте Карно имеют значение  $Y$  (нуль или единица), что означает несущественность рассматриваемой входной переменной  $X_i$ . Однако в карте состояний функции могут существовать координаты, имеющие нулевое и единичное значение для переменной  $X_i$  соответственно с противоположным значением функций  $\hat{Y}$ , что доказывает возможность многомерной активизации несущественной переменной. Таким образом, проектирование одномерного пути активизации есть способ определения избыточности переменных и элементов в схеме, соответствующей СДНФ.

**Следствие 6.** Если все переменные в примитиве избыточны или несущественны, он является несущественным. Естественно, что такого элемента не может быть в СДНФ, за исключением случая, когда

функция тождественно равна нулю или единице. В противном случае речь идет о продукте минимизации булевой функции, т. е. о ДНФ, скобочных формах и так далее.

Определение 4. Дизъюнктивная нормальная форма называется простой (ПДНФ), если в процессе минимизации каждый терм, соответствующий конституенте единицы, был включен в одно склеивание. В противном случае ДНФ назовем избыточной.

**Теорема 6.** Простая дизъюнктивная нормальная форма не содержит несущественных переменных в термах, полученных в процессе минимизации.

Минимизация есть процесс исключения из термов несущественных переменных. Это значит, что после упомянутой процедуры все соседние конституенты единицы, число которых определяется степенью двойки, будут включены в простой терм. Кроме того, изменение любой существенной для склеенного поля переменной  $X_i$  должно приводить к изменению функции, в противном случае склеивание осуществлено неполностью и его нужно продолжить, чтобы убрать несущественность переменной  $X_i$ .

**Следствие 7.** Минимизация булевой функции в целях получения ПДНФ позволяет построить полный минимальный проверяющий тест для ОКН только одномерными путями активизации.

В самом деле, если все переменные в термах существенны, то каждая из них создает собственное изменение функции, в котором не участвуют другие термы, поскольку все они формируют неперекрывающиеся состояния выходов на карте Карно. Если бы на паре входных наборов существовала другая активная переменная, то это свидетельствовало бы о несущественности переменной, что невозможно в силу условий теоремы 6.

**Следствие 8.** Если для входной переменной примитива в схеме, соответствующей ДНФ, невозможно построить одномерный путь активизации, она является несущественной (рис.2).

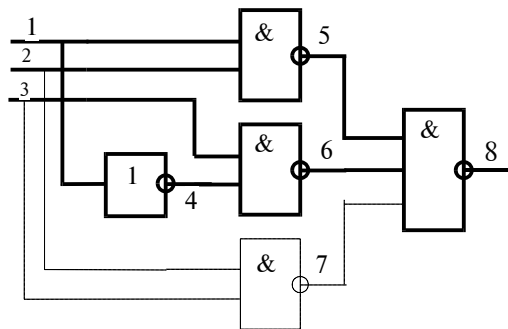


Рис. 2. Схема, соответствующая ДНФ

Выделенные элементы соответствуют простой ДНФ. Отсюда следует, что все переменные в простых термах являются существенными. Подтверждением тому могут служить пары наборов: 111-101, 110-010 и 000-001, 010-110, активизирующие входы первого и второго примитивов соответственно.

Использование избыточного терма приводит к появлению избыточного элемента, заданного на рис. 2 тонкими линиями. Данный ПЭ имеет несущественные переменные, которые не могут быть активизированы без одновременного очувствления входов других примитивов. Поскольку все входы последнего элемента несущественны, его можно и нужно убрать из схемы, что не только минимизирует

аппаратурные затраты, но и повышает тестопригодность устройства.

Другой пример избыточности, заданный функцией  $Y = X_1 X_2 \vee X_2$ , свидетельствует о несущественности переменной  $X_1$ , которую невозможно активизировать без очувствления  $X_2$ . Однако неизбыточная линия  $X_2$  активизируется вполне самостоятельно в масштабе избыточной схемы или после удаления переменной  $X_1$ , когда функция превращается в выражение  $Y=X_2$ .

**Теорема 7.** Для неизбыточной КС произвольной структуры существует одномерный путь активизации, проходящий через любой вход примитива.

Возможность построения в прямой фазе активного одномерного пути обеспечивается существенностью переменных в примитивах схемы, поэтому всегда может быть построен путь от  $X_i$  до  $Y$ . При выполнении обратного хода, в случае несущественности  $X_i$ , наблюдается фактический разрыв между  $Y$  и  $X_i$ . Однако при наличии хотя бы одной существенной переменной  $X_j$  для  $Y$  обратная импликация должна обеспечить изменение выхода  $Y$  на паре входных последовательностей. Если этого не произойдет, все входы схемы являются несущественными, что возможно при функции, тождественно равной единице или нулю на всех входных наборах. Получение положительного решения - построения пути активизации к другому входу  $X_j$ , не равному  $X_i$ , обусловлено обратной импликацией символов обеспечения  $\{O, I, A, B\}$ , которые содержат в том числе символы активности  $\{H, E, E, H\}$  соответственно. Последние могут быть получены на линии разветвления, принадлежащей сходящемуся разветвлению. Это связано с тем, что получение  $\{H, E\}$  из символов  $\{O, I, A, B\}$  есть результат пересечения пар  $A, I$  и  $O, B$  и ничего более. Поэтому отсутствие разветвлений в схеме - это гарантия ее неизбыточности, и наоборот, если схема избыточна (присутствие несущественных переменных или дублирующих элементов), то она имеет разветвления. Несущественность переменной или избыточность элемента есть свойство структуры, поэтому ее выявление возможно только на обратном ходе, где проверяется непротиворечивость гальванических соединений требующему условию активизации, полученной при выполнении прямой фазы. Однако наличие сходящихся разветвлений не является условием достаточности для определения факта схемной избыточности, поскольку при этом возможна ситуация существенности всех переменных и отсутствия избыточных примитивов.

Любая неизбыточная структура при наличии изменения в одном разряде на паре входных векторов может создавать изменение на выходе схемы. Для схемы, соответствующей ПДНФ, все переменные термов являются существенными, что означает возможность активизации любой из них с помощью одномерного логического пути. Отсюда следует, что и через входы элемента, не являющиеся внешними линиями схемы, можно построить одномерные пути активизации до внешнего выхода объекта.

Избыточность есть свойство структурной реализации функции, она всегда устранима и определяется несущественностью переменной, которая проявляется в виде активизации другой входной переменной при выполнении обратной фазы или избыточными термами. Для последнего случая характерна симмет-

ричная или несимметричная избыточность. В первом случае более двух одинаковых термов дублируют друг друга при формировании значения функции, что создает для каждой переменной мерность активизации, равную общему числу термов. Примером может служить схемная реализация функции  $Y = X_1X_2 \vee X_1X_2$ , где каждая переменная в обоих термах активизируется одновременно. Во втором случае при существовании одного или нескольких избыточных термов, не равных неизбыточным, получается одномерная активизация  $X_i$  в неизбыточном терме и многомерная (двумерная) активизация указанной переменной при первоначальной обработке  $X_i$  в избыточном терме. Например, для схемной реализации функции  $Y = X_1X_3 \vee \bar{X}_1X_2 \vee X_2X_3$  переменные в третьем терме активизируются совместно с аналогами в первых двух.

**Теорема 8.** *Для избыточной КС произвольной структуры существует путь активизации, который может быть построен с помощью К-алгоритма, проходящий через любой вход примитива.*

Поскольку в примитивах схемы отсутствует избыточность, то через любой его вход можно построить путь активизации при выполнении прямой фазы. Структурная реализация функции может иметь несущественность терма или переменной. Если вход примитива является несущественным, то на обратной фазе согласно доказательству теоремы 7 будет построен дополнительный путь активизации для фактического очувствления выхода функции. Формально работа К-алгоритма завершится построением двумерного для выбранного входа терма пути активизации. Если вход примитива принадлежит избыточному элементу, то на обратной фазе будет подтверждена активизация эквипотенциальной линии, которая идентифицирует выбранный вход терма. Однако фактическое очувствление линии осуществится через терм, отличный от избыточного. В совокупности результатом работы К-алгоритма будет двумерный путь активизации, проходящий через избыточный и неизбыточный примитивы. Для симметричной избыточности схемной реализации функции мерность пути активизации определяется количеством ПЭ симметричной избыточности.

## 7. К-алгоритм генерации тестов

Речь идет о практическом применении кубического исчисления на основе приведенных выше теоретических обоснований для построения проверяющих тестов методом очувствления одномерного логического пути эквипотенциальных линий, в котором присутствуют прямая и обратная фазы для выполнения процедур активизации и обеспечения соответственно.

Вместо двухтактных символов активизации обеспечения используются их одноктактные составляющие, реализующие активность, или обеспечение в двух соседних временных фреймах. Такое упрощение позволяет значительно повысить технологичность алгоритма генерации тестов, сведя все процедуры к выполнению операций над символами алгебры Кантора  $\{0, 1, X, \emptyset\}$ .

В качестве исходной информации используется структурно-функциональная модель устройства, где для каждого примитива имеется кубическое покрытие

$C = \{C_1, C_2, \dots, C_i, \dots, C_n\}$ , по которому генерируются пары кубов покрытия активизации. Учитывая, что символы алфавита в парах кубов активизации — одноктактные, для выполнения обратной фазы — обеспечения используются только кубические покрытия примитивов. Результат выполнения алгоритма оформляется во множество пар векторов активизации

$$D = \{D_1, D_2, \dots, D_i, \dots, D_n\}, D_i = (D_i^1, D_i^2).$$

Для исключения итеративности процедур прямой и обратной импликации до начала работы алгоритма следует выполнить ранжирование линий и элементов. Первыми нумеруются в порядке возрастания входные линии схемы, после чего очередные номера присваиваются выходам тех элементов, входы которых уже занумерованы. Последними отмечаются выходы ПЭ, относящиеся к внешним выходам схемы. Аналогично осуществляется ранжирование ПЭ, с той лишь разницей, что на первом шаге нумеруются элементы, входы которых являются внешними входами объекта, а затем отмечается очередной ПЭ, имеющий уже отмеченных предшественников. Последними нумеруются элементы, выходы которых принадлежат к внешнему разъему устройства.

С учетом приведенных определений и доказательств основные моменты К-алгоритма представлены следующими пунктами:

1. Выбор очередной входной эквипотенциальной линии  $j$  и установка в соответствующей координате нового вектора активизации символа активности (здесь и далее имеется в виду пара символов одноктактного алфавита)  $D_j = 01(10)$  (на первом шаге  $i-1, j=1$ ). Все другие координаты вектора  $D$ , за исключением  $D_j$ , к началу построения активного пути должны иметь значения неопределенностей (XX).

2. Определение очередного примитива для выполнения прямой импликации символа активизации по правилу: линия, соответствующая входу ПЭ, должна иметь символ активности в текущем векторе  $D_i$ , выходная координата элемента по возможности не должна быть активизирована во всех предшествующих векторах  $D$ . Анализ примитивов заканчивается при достижении активности на внешних выходах схемы. Стратегия выбора очередного пути активизации использует критерий получения минимального множества одномерных путей, покрывающих все эквипотенциальные линии объекта.

3. Обратная фаза заключается в обработке каждого примитива схемы, на выходе которого после прямой фазы существуют символы, отличные от (XX), исключая элементы, ставшие активными при выполнении прямой фазы. Расширение поля деятельности обратной импликации в целях доопределения символов активизации  $01(10)$  связано с построением фактического пути, очувствляющего выход схемы, если первоначально, при выполнении прямой фазы была сделана попытка активизировать несущественную переменную. Для устранения итеративного характера обратной импликации очередным следует выбирать элемент, имеющий уже обработанных предшественников. Суть анализа заключается в пересечении каждого куба  $C_i$  покрытия обеспечения с текущим вектором активности  $D_i$ . При непустом пересечении  $D_i$  с  $n$  кубами выполняется размножение

строки  $D_i$  на  $n$  векторов, что является следствием многовариантности решения задачи обратной импликации.

4. Объединение текущих векторов, являющихся претендентами для активизации очередного входа устройства, после выполнения обратной импликации на очередном примитиве. Данная процедура имеет смысл при наличии сходящихся разветвлений. Она способна значительно количество промежуточных решений, а значит и уменьшить время работы алгоритма. Объединение выполняется в случае различия пары строк только по одной координате – основной принцип минимизации, или при получении в результате выполнения координатной операции пересечения двух соседних векторов одного из них – правило поглощения.

5. После проектирования пар векторов активизации, покрывающих все линии в схеме, выполняется доопределение символа  $X$  на каждой входной координате по правилу

$$D_{ij}^{1(2)} = D_{ij}^1 \cap D_{ij}^2 \Leftarrow (D_{ij}^1 \vee D_{ij}^2 = X) \wedge (D_{ij}^1 \neq D_{ij}^2).$$

В противном случае доопределение подчинено правилу

$$(D_{ij}^1 \cap D_{ij}^2 = X) \Rightarrow [D_{ij}^1 = D_{ij}^2 = 0(1)].$$

Координаты заменяются нулем или единицей в целях получения минимального кодового расстояния между соседними векторами теста. Возможны и другие критерии доопределения, например, с учетом максимизации проверяющих свойств двоичных наборов или минимизации входных векторов вычеркиванием одинаковых.

6. Для определения двоичных значений линий на всех тестовых последовательностях выполняется их моделирование, которое уточняет состояния невходных линий, оставшихся неопределенными даже после выполнения обратной фазы. Необходимость выполнения данного пункта иллюстрируется состояниями линий фрагмента схемы, изображенной на рис.3.

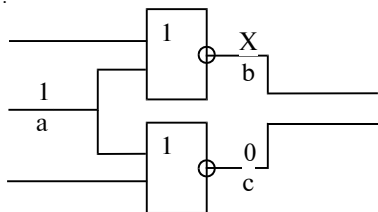


Рис. 3. Фрагмент схемы

Полученный в результате выполнения  $K$ -алгоритма тест является полным относительно ОКН, если удалось построить только одномерные пути активизации, покрывающие все линии. В противном случае тест будет полным относительно всех ОКН всех избыточных или существенных линий. Минимальность полученного двоичного теста не гарантирована, поэтому есть смысл в оценке его качества с помощью программ моделирования неисправностей, что дает возможность получить минимальный набор входных проверяющих последовательностей.

Иллюстрацию основных шагов алгоритма выполним на примере обработки схемы, представленной на рис. 2, покрытия элементов которой записаны по входным и выходным номерам эквипотенциальных линий:

$$C^1 = \begin{pmatrix} 1 & 2 & 5 \\ 3 & 4 & 6 \\ 3 & 2 & 7 \end{pmatrix}; C^2 = \begin{pmatrix} 1 & 4 \\ 0 & 1 \\ 1 & 0 \end{pmatrix}; C^3 = \begin{pmatrix} 5 & 6 & 7 & 8 \\ 0 & X & X & 1 \\ X & 0 & X & 1 \\ X & X & 0 & 1 \\ 1 & 1 & 1 & 0 \end{pmatrix}.$$

Покрывтия пар кубов активизации упомянутых типов примитивов схемы имеют вид:

$$C^1 = \begin{pmatrix} 1 & 2 & 5 \\ 3 & 4 & 6 \\ 3 & 2 & 7 \end{pmatrix}; C^2 = \begin{pmatrix} 1 & 4 \\ 0 & 1 \\ 1 & 0 \end{pmatrix}; C^3 = \begin{pmatrix} 5 & 6 & 7 & 8 \\ 0 & X & X & 1 \\ 1 & 1 & 1 & 0 \\ X & 0 & X & 1 \\ 1 & 1 & 1 & 0 \\ X & X & 0 & 1 \\ 1 & 1 & 1 & 0 \end{pmatrix}.$$

Для схемы (рис.2) выполнение прямой фазы определяет следующие векторы активизации:

$$D = \begin{pmatrix} 1 & 2 & 3 & 4 & 5 & 6 & 7 & 8 \\ 1 & 1 & - & - & 0 & X & X & 1 \\ 0 & X & - & - & 1 & 1 & 1 & 0 \\ 1 & - & 1 & 0 & 1 & 1 & 1 & 0 \\ 0 & - & X & 1 & X & 0 & X & 1 \\ 1 & 1 & - & - & 0 & X & X & 1 \\ X & 0 & - & - & 1 & 1 & 1 & 0 \\ - & 1 & 1 & - & X & X & 0 & 1 \\ - & 0 & X & - & 1 & 1 & 1 & 0 \\ - & - & 1 & 1 & X & 0 & X & 1 \\ - & - & 0 & X & 1 & 1 & 1 & 0 \\ - & 1 & 1 & - & X & X & 0 & 1 \\ - & X & 0 & - & 1 & 1 & 1 & 0 \end{pmatrix},$$

где знак “-” соответствует символу  $X$ . Выполнение обратной фазы для доопределения символов  $D_{ij}^1(D_{ij}^2)$

при выполнении условий  $D_{ij}^2(D_{ij}^1) = X \wedge (D_{ij}^1 \neq D_{ij}^2)$  на невходных линиях путем использования кубических покрытий задает тест активизации:

$$D = \begin{pmatrix} 1 & 2 & 3 & 4 & 5 & 6 & 7 & 8 \\ 1 & 1 & X & 0 & 0 & X & X & 1 \\ 0 & X & 0 & 1 & 1 & 1 & 1 & 0 \\ 1 & 0 & 1 & 0 & 1 & 1 & 1 & 0 \\ 0 & 0 & X & 1 & X & 0 & X & 1 \\ 1 & 1 & X & 0 & 0 & X & X & 1 \\ X & 0 & 0 & X & 1 & 1 & 1 & 0 \\ X & 1 & 1 & X & X & X & 0 & 1 \\ 0 & 0 & 0 & 1 & 1 & 1 & 1 & 0 \\ 0 & X & 1 & 1 & X & 0 & X & 1 \\ X & 0 & 0 & X & 1 & 1 & 1 & 0 \\ X & 1 & 1 & X & X & X & 0 & 1 \\ 0 & X & 0 & 1 & 1 & 1 & 1 & 0 \end{pmatrix}.$$

Доопределение символов  $X$  на входных координатах полученного теста в соответствии с правилами п. 5  $K$ -алгоритма задает наборы  $D^X$ , которые после минимизации, вычеркиванием одинаковых приобретают вид  $D_m^X$ :

$$D^X = \begin{pmatrix} 1 & 2 & 3 \\ 1 & 1 & 1 \\ 0 & 1 & 0 \\ 1 & 0 & 1 \\ 0 & 0 & 1 \\ 1 & 1 & 0 \\ 1 & 0 & 0 \\ 0 & 1 & 1 \\ 0 & 0 & 0 \\ 0 & 0 & 1 \\ 1 & 0 & 0 \\ 0 & 1 & 1 \\ 0 & 1 & 0 \end{pmatrix}; D_m^X = \begin{pmatrix} 1 & 2 & 3 \\ 1 & 1 & 0 \\ 0 & 1 & 0 \\ 0 & 1 & 1 \\ 1 & 0 & 0 \\ 0 & 1 & 1 \\ 0 & 0 & 0 \end{pmatrix}.$$

Полученный минимизированный тест  $D_m^X$  содержит семь наборов, а его проверяющие способности представлены в табл. 2 моделирования неисправностей.

Таблица 2

Исправное моделирование								Проверяемый дефект							
1	2	3	4	5	6	7	8	1	2	3	4	5	6	7	8
1	1	0	0	0	1	1	1	0	0	.	.	1	.	.	0
0	1	0	1	1	1	1	0	1	.	1	.	0	0	0	1
1	0	1	0	1	1	1	0	0	1	.	1	0	0	0	1
0	0	1	1	1	0	1	1	1	.	0	0	.	1	.	0
1	0	0	0	1	1	1	0	.	1	.	.	0	0	0	1
0	1	1	1	1	0	0	1	.	.	0	.	.	.	.	0
0	0	0	1	1	1	1	0	.	.	1	.	0	0	0	1

Моделирование ОКН для наборов  $D_m^X$  позволяет сделать вывод об избыточности последних трех, которые дублируют проверку неисправностей, выполненную предыдущими векторами. Поэтому минимальным двоичным тестом являются последовательности:

$$D_{\min}^X = \begin{pmatrix} 1 & 2 & 3 \\ 1 & 1 & 0 \\ 0 & 1 & 0 \\ 1 & 0 & 1 \\ 0 & 0 & 1 \end{pmatrix}.$$

Моделирование неисправностей также подтверждает невозможность проверки дефекта 7<sup>1</sup> на полученных векторах, поскольку выход 7 относится к избыточному элементу.

Выполнение К-алгоритма, основанного на активизации минимального множества путей в схеме, от внешних входов до наблюдаемых выходов, покрывающих все линии, гарантирует построение полного проверяющего теста для тех ОКН, для которых он существует. Дальнейшая минимизация теста связана с применением средств моделирования неисправностей, что дает точную характеристику проверяющих свойств каждого вектора и теста.

## 8. Заключение

Разработаны теоретические основы применения ДКИ для проектирования тестов с использованием идеи очувствления логического пути от внешнего входа до наблюдаемых выходов. Показаны преимущества А-алфавита для компактной записи кубов активизации и обеспечения, необходимых для работы предложенного кубического алгоритма, способного построить тест для произвольной структуры объекта, в том числе и избыточной, если он существует. К-алгоритм адаптирован к троичному алфавиту

кубического исчисления, что значительно упрощает процедуры прямой и обратной импликаций, являющиеся основой проектирования проверяющих тестов.

Что касается практического применения полученных результатов, то с появлением научного и практического интереса к языку VHDL [4,5] предлагаемый метод может быть использован в качестве базового для семейства цифровых схемных реализаций структурно-функционального уровня описания. Ориентация на развитие и совершенствование языковых и программных средств VHDL является следствием:

1) создания замкнутого технологического цикла проектирования и производства интегральных схем на основе VHDL;

2) появления мощных компиляторов с языка VHDL и программных средств создания и верификации проектов;

3) стандартизации языка VHDL на уровне науки и производства развитых стран;

4) открытости и доступности структур данных среды VHDL-Active для улучшения и модификации программных компонентов;

5) популяризации на рынке стран СНГ новых технологий VHDL-Active [6].

В целом следует отметить, что в настоящее время появилась реальная возможность интегрирования разработок в области САД-систем в европейскую и мировую науку посредством соединения неустаревающих теоретических достижений в области автоматизации проектирования и верификации компьютерных и микросистемных изделий с программными технологиями на основе мощных компьютерных систем, например, фирмы ALDEC. Подтверждением сказанному могут служить выводы [7], направленные на обоснование долгосрочного прогноза перспективности развития новых технологий проектирования под эгидой языка и среды VHDL.

**Литература.** 1. *Breuer M.A., Friedman A.D.* Diagnosis and reliable design of digital system. Woodlound wills, Computerscience press inc., 1976. 308p. 2. *Автоматизированное проектирование цифровых устройств / С.С.Бадулин, Ю.М.Барнаулов и др. / Под ред. С.С. Бадулина. М.: Радио и связь, 1981. 240с.* 3. *Хаханов В.И.* Техническая диагностика элементов и узлов персональных компьютеров. К.: ИЗМН. 1997. 308с. 4. *Ashenden, Peter.* Designer's Guide to VHDL. Morgan Kaufman, 1994. 500p. 5. *Thomas, D.E. & Moorby, P.* Verilog Hardware Description Language.— Kluwer Academic, 1991. 223p. 6. *Active-VHDL Series. Book 1. Reference Guide.* ALDEC Inc. 1998. 205 p. 7. *Techniques of Informatics and Microelectronics for Computer Architecture.* TИМА Laboratory. Annual Report 1998. В. Courtois. 1999. 358p.

Поступила в редколлегия 12.06.99

**Рецензент:** д-р техн. наук Кривуля Г.Ф.

**Хаханов Владимир Иванович**, д-р техн. наук, профессор кафедры автоматизации проектирования вычислительной техники ХТУРЭ. Научные интересы: техническая диагностика вычислительных устройств, систем, сетей. Увлечения: баскетбол, горные лыжи. Адрес: Украина, 310726, Харьков, пр. Ленина, 14, тел. 40-93-26.

**Сысенко Ирина Юрьевна**, студентка кафедры автоматизации проектирования вычислительной техники ХТУРЭ. Научные интересы: техническая диагностика вычислительных устройств. Хобби: иностр. языки, музыка. Адрес: Украина, 310726, Харьков, пр. Ленина, 14, тел. 40-93-26.

**Абу Занунех Халиль И.М.**, аспирант кафедры автоматизации проектирования вычислительной техники ХТУРЭ. Научные интересы: техническая диагностика вычислительных устройств. Хобби: шахматы, футбол, теннис. Адрес: Украина, 310726, Харьков, пр. Ленина, 14, тел. 40-93-26.